Appendix J

Title:
Protocol Machine Transmit Version 2.2

1 Introduction

1.1 Overview

The Protocol Machine Transmit (PT) provides channelized transmit protocol services for the following modes:

1. HDLC
2. bit-synchronous PPP
3. octet-synchronous PPP
4. Transparent

The PT architecture consists of a single logic core and a channel context memory for the channel's configuration and protocol state e.g. CRC sum, bit stuffing '1's count, interframe fill count, etc. The physcial context memory which is outside of the macro itself would be RAM when a large number of channels are supported. A register bank could be used when only a few channels are implemented. With this architecture, the core logic is essentially fixed except for channel address bus width. The application's clock speed and required aggregate throughput determine the maximum number of channels that can be supported.

Important features:

- HDLC, bit-synchronous PPP, octet-synchronous PPP, and Transparent modes
- Flexible scaling by dimensioning of context memory
- Per channel protocol configuration
- Configuration via Simplifed Microcontroller Interface (SMIF) registers
- Aggregate throughput of 52.8 MBit/second @ 33 MHz clock

1.2 System Integration

The PT has interfaces for :

- Configuration and Control : 32 bit Simplified Microcontroller Interface SMIF)
- Service Data Input : 32 bit Transmit Buffer (TB)
- Protocol Data Output ; 8 bit data port for Timeslot Assigner Transmit (TT)
- Interrupts : end-of-frame or buffer under-run

The maximum aggregate throughput of 52.8 Mbit/second @ 33Mhz system clock is attained with:

- 1 octet of protocol data transferred to TT every 5 system clocks
- 1 dword from TB per 20 clocks on average.

Interrupts are used to report failures during protocol handling e.g., TB underrun or to report the end of a transmitted frame in order to coordinate release of transmit buffers.

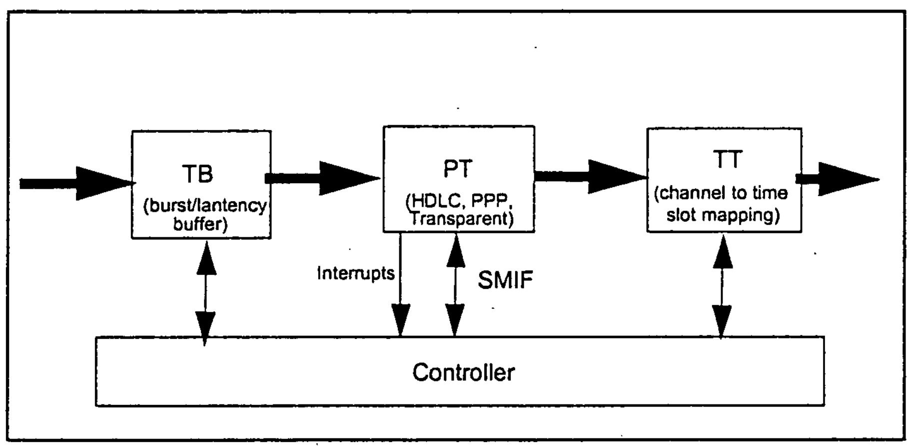

Figure 1 System Integration

1.3 Known Restrictions and Problems

2 Functional Description

2.1 Block Diagram incl. Clocking Regions

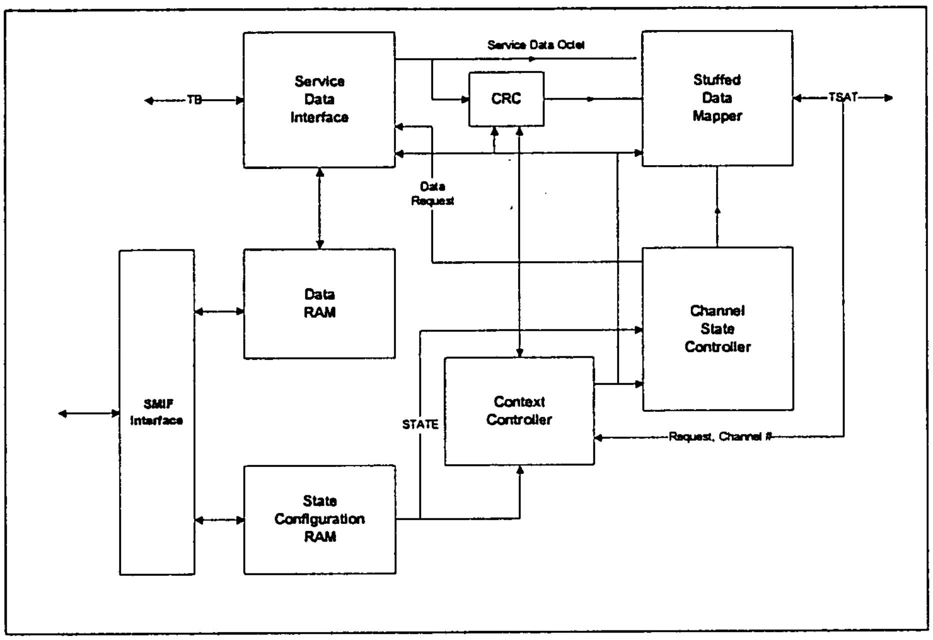

Figure 2
PT Bock Diagram

2.2 Normal Operation

2.2.1 Functions Common to all Protocols

To minimize the overhead of context switching, PT processes TT channel requests on octet sized data blocks. In order to provide flexibility to systems which use smaller blocks, the TT can also use a mask to enable individual bits in the octet. Unused bits of the octet are set to '1'. The mask essentially converts the TT interface from octet sized to N bits (0 < N <=8) per TT service. Although this adds flexibility, it should be used with caution because it reduces the aggregate. A mask is attractive for sub-channels defined in a timeslot because is greatly simplifies the TT design.

Note that TT could be replaced in an application with an interface to multiple, high or low speed serial data links by providing serial to parallel converters or converting the interface to bit serial operation via the mask. The channel number provided to PT would then correspond to link number. For historical reasons, Time Slot Assigner Transmit (TT) will continue to be used in this document although it is somewhat limiting in scope.

A pipelined design is used within PT to improve logic timing. Because PT channel processing is only performed in response to TT requests, the pipeline delays the start of a channel by several TT request cycels. After a channel is initialized via SMIF iterface, it remains in an idle state with all PT pipelines empty until the first TT request. PT responds as follows:

1. Load channel context into PT core logic.
2. Transfer protocol data in pipeline immediately to TT in bits specified by octet mask. If no protocol data available, transfer idle code or interframe fill or abort when open frame (under-run). Genererate Interrupt if under-run and the interrupt is enabled.
3. Return ready to TT.
4. If service data from TB is available in the PT's input buffer and the protocol pipeline has sufficient free space, generate new protocol data for the channel just serviced.
5. Save channel state back to context memory.
6. If PT's input buffer has free space for a double word, request data from TB.

PT provides the active status of a channel by asserting PT_TTCH_ON when it returns data to TT. Use of this status by TT is application specific e.g. TT might drive its outputs into tri-state for an uninitialized channel or it might simply ignore it and use data from PT transparently.

Protocol data for TT be be inverted on a per channel basis. This is useful for HDLC protocols over AMI links which cannot tolerate long strings of zeros. It converts the '0' stuffing of HDLC into '1' stuffing.

Status words are used to transfer data between the TB and PT when:

1. dword contains < 4 octets of service data or
2. end of frame for HDLC or PPP frame (FE=1) or
3. abort of open HDLC or PPP frame required (TAB=1).

A status word with BE=0 (no data) and neither flag set is invalid. The TBSTAT line is asserted high along with TBRDY=1 to indicate a status word.

Table 1 Status word

Bit	31	30	29..26	25	24	23..	..16	15..	..8	7..	..0
Function	FE	TAB	Reserved	BE		Byte2		Byte1		Byte 0	

If BE = 00 , the dword contains only status information. Otherwise BE is the number of valid service data octets starting at Byte 0.

TAB=1 causes an abort sequence to be sent if a frame is open. Otherwise it is ignored.

FE=1 indicates end of frame. FCS will be inserted followed by a closing flag.

2.2.2 HDLC mode

PT performs only the low level functions of frame generation i.e. flag insertion, CRC generation, and bit stuffing.

Flag	Address	Control	Information	CRC	Flag
0111 1110	8 bits	8 bits	<=0 Bits	16/32 bits	0111 1110

Figure 3 HDLC Frame Format

The frame start and end is marked with the flag character (7Eh). Interframe time fill character of either 7Eh or FFh is sent when a frame is not open. The mimimun number of interframe characters to be transmitted is controlled by the parameter FNUM (0<=FNUM<=255). The use of FNUM is:

FNUM = 0 : shared closing and opening flag !

FNUM /= 0 : if interframe fill character is 7Eh (flag),

# of interframe characters = (1+ FNUM) flags

else (interframe fill character is FFh)

# of interframe characters = 2flags + FNUM 0FF hex characters

A shared 0 bit between two flags is not supported in the PT.

When PT reads service data from TB, a new frame is automatically started after the programmed number of interframe fill characters have been sent. **Note that FNUM is not used at the initial start of a channel i.e. the first frame is sent without delay.** The Frame Check Sequence (CRC16 or CRC32) is initialized at the beginning of the frame and then updated for all service data in the frame. Zero bit insertion (stuffing) is

performed by inserting a '0' bit after all sequences of five contiguous '1' bits in the service data and the final FCS.

If an underrun occurs in TB, it asserts the TB_EMPTY line along with TBRDY at the end of a PT read cycle. When this happens in the middle of a frame, an abort sequence "01111111" is automatically generated. An underrun interrupt is generated when it is enabled.

The abort sequence is also generated but without an interrupt when the Transmit Abort falg TAB is set in the status word. This flag would be used when removing a channel from service or interrupting an open frame. An TAB is ignored if a frame is not open.

When the frame end is reached (FE=1 in the status word), the FCS is transferred to TT after bit stuffing, the closing flag inserted, and the interframe fill mode entered. A new frame can be started as soon as the interframe fill count is satisfied.

2.2.3 Bit synchronous PPP

The mode is identical to the HDLC mode except for the abort sequence when the interframe fill pattern is 0FF hex. When 7Eh is programmed as interframe time fill character, the abort sequence consists of "01111111" as in HDLC. When 0FFh is programmed, the abort sequence consists of 15 '1's.

2.2.4 Octet Synchronous PPP (OSPPP)

OSPPP uses a frame structure similar to the HDLC mode:

- frame start and end synchronization is performed using the flag character (7Eh).
- 16 or 32 bit CRC is computed over all service data read from TB and appended to the end of the frame.
- interframe fill character is used but will always be 7E hex (flag). the use of FNUM is identical to HDLC case for 7E fill pattern.

OSPPP differs by using octet stuffing of service data and FCS instead of '0' bit stuffing. Octet stuffing replaces certain specified octet values with a 2 octet sequence consisting of $7D_{Hex}$ (Control Escape) followed by the octet EXOR'ed with 20_{Hex} (e.g. 13 Hex is mapped to 7Dh, 33h). OSPPP allows suppression of special control characters used by transmission equipment such as modems. These characters can be transparently inserted because they can be recognized as spurious and removed at the end point OSPPP handler since valid data will be substituted. **Note that this mode can only be used on links which are character oriented.** Otherwise, adjacent characters could inadvertently form a flag in the bit sequence e.g. 57 + E9 hex This precludes use of OSPPP on sub-channels (mask /= 0FF hex) because the characters can't be synchronized except by the time slot boundary.

Characters which are stuffed are :

- Control Escape $7D_{Hex}$
- $7E_{Hex}$ (flag character) in service data or FCS (not opening or closing flags)

- DEL control character (optional)
- characters specified in 32 bit Asynchronous Control Character Map (ACCM)(optional)
- characters specified in 4 bit Extended ACCM (EACCM)(optional)

The substitution of Control Escape and flag characters is mandatory and so doesn't require enabling flags. The ACCM, EACCM, and DEL enables are per channel. The ACCM enables stuffing for all characters in the range 00-1F_{Hex} . EACCM enables stuffing of 4 ACCM extension characters which are specified characters in a user programmable 32 bit (4 character) register common to all channels. The opening and closing flags are not stuffed.

The OSPPP abort sequence consists of the Control Escape character 7D hex followed by a flag character 7E_{Hex} (not stuffed). Aborts are generated identically to HDLC cases.

Between two frames, the interframe time fill character is always 7Eh. The count of interframe fill characters is set by FNUM as described in the HDLC description.

2.2.5 Transparent Mode (TM)

In the transparent mode, PT performs data transmission without any framing, i.e. without

- Flag insertion
- CRC generation
- Bit or octet stuffing

The data read from TB is simply buffered in PT and copied to TT after the channel is synchronized. Synchronization is provided to support super-channels (fractional T1 or T1) for TM. The start of the transmission of data is delayed by PT until the time marker TT_SYNC is asserted by TT at the start of its read cycle. Normally this would occur at the first time slot of a superchannel. After initialization in TM, PT remains in a waiting state where it transfers all ones data to TT. When TT_SYNC line is asserted and the PT pipeline has protocol data, the channel is synchronized and begins transferring data.

After the first activation of TT_SYNC, PT ignores TT_SYNC and transfers data to TT until an underrun occurs. TM does not have an interframe time i.e. once the channel is synchronized, it must always receive data from TB without interruption. A Frame End flag is not valid. Underrun causes PT to go back to the waiting state until TT_SYNC is again asserted. An underrun generates an interrupt when enabled.

A programmable TM Flag (TFLAG) character can be sent whenever service data is not being transferred (TM interframe fill character).

An 'Transparent Mode Pack' option is provided to support subchanneling. When subchanneling is used (logical channels of less than 64 kbit/s), service data is delivered only in bit positions marked by a '1' in the mask from TT. Unused bits are fixed at '1'. The user would normally send packed data to PT maps it into the enabled bit positions and inserts '1' bits in bit positions not enabled. If the mode is not packed, the user service data is transferred to TT regardless of the mask. The assumption is that the user has

already unpacked the data and inserted the '1' bits at unused bit positions. This mode is not recommended and provided only for compatibility reasons.

2.3 Configuration and State RAMs

The following tables describes the configuration and state RAMS used by the PT

Table 2 Channel Configuration RAM layout (CN-1 words x 59 bits)

Bit / Field Name	# Bits	Description
ACCM	32	- 32 bit Async Character Control Map for OSPPP Mode - 8 bit TFLAG for Transparent Mode (lower 8 bits)
CH_EN	1	Enable channel
MODE	2	Protocol Mode (HDLC, Bit PPP, Octet PPP, Transparent)
CRC_DIS	1	CRC disabled
CRC32	1	CRC mode (0 -> CRC16, 1->CRC32)
TMP	1	Transparent Mode Pack 0 -> data octet transferred independent of enable mask 1 -> data octet unpacked as per enable mask (normal)
INVERT	1	Invert protocol data to TT
FA	1	Transparent Mode Flag Adjust 0 -> send FF in exceptions state (idle/waiting for sync) 1 -> send user specified TFLAG in exception state
IFTC	1	Interframe Time Character (HDLC or Bit Oriented PPP) 0 -> 7E hex 1 -> FF hex
ACCM_EXT	4	Extended ACCM map for user characters (OSPPP)
MAP_DEL	1	Enable DEL character octet stuff (OSPPP)
DIS_FE_INT	1	Disable Frame End interrupt
DIS_UR_INT	1	Disable Underrun Interrupt
INT_QUEUE_ID	3	Channel Interrupt Queue Identifier
FNUM	8	Interframe fill count (0-255)

Channel State RAM (CN-1 words x 105 bits)

Bit / Field Name	# Bits	Description
ROCTS	32	Remaining Octets in service wordto be translated
CRC	32	Frame Check Sequence being generated for open frame

Channel State RAM (CN-1 words x 105 bits)

Bit / Field Name	# Bits	Description
RBITS	17	Remainding protocol bits after transfer to TT: At least 8 bits must be available to immediately service TT on demand. When 7 remain, protocol data is generated at the end of TT service. For HDLC, 8 bit octet can be stuffed twice => 7+8+2 = 17 bits required.
RBITS_CNT	5	Number of valid RBITS
ONES_COUNT	3	Number of contiguous '1' s (0-4) for HDLC stuffing, flag, or abort detection.
STATE	3	Channel state
ROCNT	3	Number of valid ROCTS (0-4)
STAT	1	STAT flag for ROCTS context
NO_DS	1	flag indicating no data or status available in context
DLY_CH_ON	1	Pipelined (delayed) flag to turn on/off the channel
CCNT	8	Control Counter (dual use) a. control insertion of CRC octets at frame end b. count interframe fill characters
PPP_CTRL	1	OSPPP octet stuffing flag

2.4 Reset Behavior

All internal registers and functions are asynchronously reset to defined states. The RESET signal must be removed synchronously to ensure reliable operation.

After reset the reception of data is turned off for all the channels and all registers are accessible via SMIF registers for initialization.

3 Macro Interfaces and Signal Description

All signals are active high until otherwise specified. Active low signals are designated by "_N" appended to their names. To make the design as re-usable as possible, bus signals with application dependent width are specified with one of the following parameters:

Parameter name	Bus Type
CN	Channel Number Bus
DB	Data/Status Bus

3.1 Signal Description

Table 3
Interfaces and Signal Description

Symbol name	I/O	Function
Clock and Reset		
SYSCLK	I	Clock
RESET_N	I	General reset of PT. All registers and RAM reset
STOP	I	Blocks PT from processing new TT requests
PT_IIP	O	PT initialization active. Writing all RAMs to defined state
Timeslot Assigner Transmit (TT) Interface		
TTRD_N	I	TT data request.
TTCH[CN-1:0]	I	channel number
TTMASK[7:0]	I	enable mask. mask bit N enables bit N in PT_TTD '0' => PT_TTD[N] not used. PT_TTD[N] = '1' '1' =>PT_TTD[N] is protocol data
TTSYNC	I	channel synchronization line (Transparent Mode only)
PT_TTRDY	O	PT will be able to finish transaction in current clock cycle

Table 3
Interfaces and Signal Description (cont'd)

Symbol name	I/O	Function
PT_TTCH_ON	O	Channel Operating Status 0 => channel is off. PT_TTD is not valid. 1 => channel is on. PT_TTD is valid.
PT_TTD[7:0]	O	Protocol data octet

Transmit Buffer (TB) interface

PT_TBRD_N	O	Request TB read
PT_TBA[CN-1:0]	O	Channel Number for TB read request
TBRDY	I	TB completing read cycle this clock
TBD[DB-1:0]	I	Read data. Valid when TBRDY='1'
TBSTAT	I	TBD data type '0' => TBD is DB/8 protocol data octets '1' => TBD is status word . TBSTAT is valid when TBRDY='1'
TBEMPTY	I	TB empty flag '0' => TBD is valid '1' -> No data or status available for channel PT_TBA. TBEMPTY is valid when TBRDY='1'

Interrupt Interface

PT_URUN_INTRPT	O	Interrupt :TB Underrun during open frame (error)
PT_FE_INTRPT	O	Interrupt : Frame End (not error)
PT_INTRPT_CN[CN-1:0]	O	Channel number
PT_INTRPT_QID[3]	O	Interrupt Queue ID Per channel user specified parameter
INT_ACK	I	Interrupt service acknowlegement

SMIF Interface

Table 3
Interfaces and Signal Description (cont'd)

Symbol name	I/O	Function
BPI_RD_SFR_N[7:1]	I	Read special function registers (1 select per register) **Bit Function** 1 : channel mode 2 : channel Asynchronous Control Character Map 3 : Extended Asynchronous Control Characters 4 : channel interrupt priority (queue ID) 5 : channel interrupt masks 6 : test access command register 7 : test access data register Note that bit select 0 is not implemented because command register is write only.
BPI_WR_SFR_N[7:0]	I	Write special function registers (1 select per register) **Bit Function** 0 : channel command register 1 : channel mode 2 : channel Asynchronous Control Character Map 3 : Extended Asynchronous Control Characters 4 : channel interrupt priority (queue ID) 5 : channel interrupt masks 6 : test access command register 7 : test access data register
BPI_REQ_N	I	BPI Request '0' => BPI_RD_SFR_N or BPI_WR_SFR_N valid '1' => BPI_RD_SFR_N or BPI_WR_SFR_N not valid
BPI_DI[DB-1:0]	I	SMIF write data input bus
BPI_RDY_N	O	SMIF read/write cycle ending this clock
BPI_D_O[DB-1:0]	O	SMIF read data output bus

3.2 Data Flow and Functional Timing

3.2.1 Timeslot Assigner (TT) Interface

The transfer is initiated by TT by asserting TTRD_N='0' along with channel number and mask lines. PT returns protocol data when the PT_TTRDY line is active.

TT must also activate the TTSYNC line during the first timeslot of a superchannel.

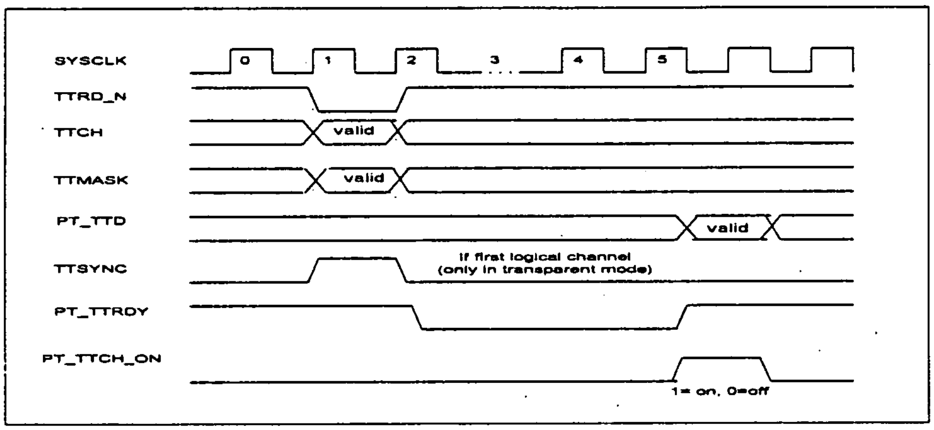

Figure 4
Data transfer from PT to TT

3.2.2 TB Interface

PT initiates an address cycle by asserting channel address PT_TBA and read request PT_TBRD_N. TB must capture the channel number from the address bus during this cycle. During the data phase, TB asserts TBRDY, TBSTAT, TBEMPTY, and TBD at the beginning of the clock cycle that the data is available. TB may insert wait states but the response time must be appropriate to the application. **A maximum of 4 wait states is allowed in any application. If TB uses 4 wait states, it must be capable of accepting a new read cycle simultaneously with the assertion of its TBRDY.**

TBEMPTY is used to stop a PT transfer when the TB has no service data for the addressed channel. TB terminates the cycle normally with TBRDY but asserts TBEMPTY to show that the data is not valid.

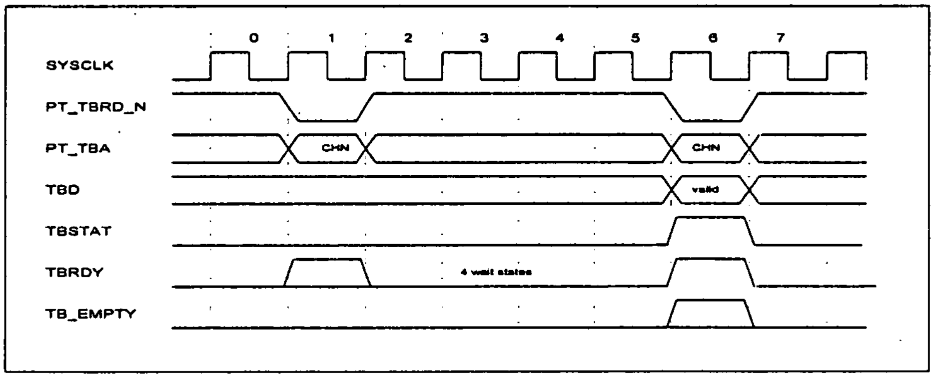

Figure 5 PT read cycles: 1 success and 1 interrupted transfer

4 Special Function Registers Description

4.1 Channel Command Register

31															16
CMD_XMIT(7:0)								0	0	0	0	0	0	0	0

15							8	7							0
0	0	0	0	0	0	0	0	CHANNAL							

CHANNAL specifies the channel number to be programmed or to be read.

The Channel Command Register can be used in a "broadcast" environment where a command may be written to more than 1 macro and each macro responds appropriately and in parallel. The actual binary value for each command is set by parameters in the VHDL package at synthesis time.

The commands supported by PT are:

1. Initialize

   Copies channel parameters from SMIF registers into context memory.
   Activates channel.

2. Channel Off

   Deactivates channel after channel's protocol data is flushed from PT context memory.

3. Initialize/Update

   For PT, identical to Initialize command. Update PT parameters after initialization. Provided for systems where Initialize command shared by other macros that cannot update their parameters.

4. Transmit Debug

   Loads channel number for subsequent read of channel parameters.

5. Send Idle

   Deactivates channel if it is on. Send idle code specified in FNUM parameter of configuration data.

For any other command, PT returns ready but ignores the command.

4.1.1 Channel Configuration

Access : read/write

Reset Value : 00000000_H

31	
FNUM(7:0)	TFLAG(7:0)

15											0
IFTF	0	FA	INV	TMP	0	CRC 32	CRC DIS	EACCM (3:0)	DEL	0	PMD(1:0)

PMD(1:0): Channel Mode

00 : HDLC

01 : BSPPP (bit synchronized)

10 : OSPPP (octet synchronized)

11 : Transparent

DEL: Del Character map enable flag (OSPPP mode only)

EACCM : Extended ACCM for 4 user specified characters (OSPPP mode only)

0 => disables mapping of respective character in EACC register
1 => enables mapping of respective character in EACC register

CRCDIS: Disable Frame Check Sequence (CRC) for HDLC or PPP

0 => CRC is inserted at end of frame
1 => CRC is not inserted at end of frame

CRC32: Enable CRC32 FCS

0 => 16 bit CRC is enabled
1 => 32 CRC is enabled

TMP: Transparent Mode Pack mode (transparent mode only)

0 => TT mask OR'ed with service data octet (no unpacking by PT)
1 => service data unpacked by PT and distributed according to TT mask.

INV: Invert channel protocol data (idle pattern when channel is off not affected)

0 => no inversion

1=> invert channel' s transmit protocol data

FA: Transparent Mode fill pattern (exception conditions)

0 => insert 0FF hex when channel is idle or not synchronized
1 => insert TFLAG when channel is idle or not synchronized

IFTF: Interframe Time Fill Character (HDLC or BSPPP only)

0 => $7E_H$
1 => FF_H

TFLAG : Transparent Mode Flag (Exception Condition Fill Pattern only if FA=1)

8-bit "fill flag" for transparent mode. It is inserted into transmit data when FA=1 and channel is idle or not synchronized in transparent mode.

FNUM : Minimum number (0-255) of interframe fill characters (HDLC or PPP only).

4.1.2 Channel Asynchronous Control Character Map (ACCM)

Access : read/write

Reset Value : 00000000_H

31 16

1F	1E	1D	1C	1B	1A	19	18	17	16	15	14	13	12	11	10

15 0

0F	0E	0D	0C	0B	0A	09	08	07	06	05	04	03	02	01	00

Used in octet synchronous PPP mode only. When bit N is set in the map, character with value N (00 to 1F) replaced by Control ESC sequence in the outgoing data stream.

4.1.3 Extended Asynchronous Control Characters

Access : read/write

Reset Value : 00000000_H

31	16
CHAR3(7:0)	CHAR2(7:0)

15	0
CHAR1(7:0)	CHAR0(7:0)

This register is common to all octet synchronous PPP mode channels. When enable flag is set in the channel's XACCM (4 bits in Channel Configuration), the corresponding character written in this register will be replaced with a Control Escape sequence. This extends the basic 32 character ACCM with 4 user specified characters.

4.1.4 Channel Interrupt Priority

Access : read/write

Reset Value : 00000000_H

31	
TQUEUE(2:0)	not used.

15	0
not used	

TQUEUE : channel interrupt priority

PT does not use this parameter directly but stores it and provides it when it generates an interrupt . The interrupt controller can use the parameter to prioritize the interrupt.

4.1.5 Channel Interrupt Masks

Access : read/write

Reset Value : 00000000_H

31								23	22						16
0	0	0	0	0	0	0	0	UE	FE	0	0	0	0	0	0

15															0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Interrupt masks disable their respective interrupt when set high.

FE : disable Frame End interrupt

UE : disable channel data Under-run Error

4.1.6 Test Command Register (TAC)

Access	: write
Reset Value	: 00000000_H

31

MID	0	0	0	AI	CMD

15 0

ADDRESS

MID:	Macro ID Code
AI:	Auto Increment Function
Address:	RAM read/write address bits
CMD:	Command

The test access provides read/write access of PT state memory. Because the state bus is 105 bits (refer to previous state description), 4 commands are implemented to read it in SMIF sized slices. PT accepts a test command only when the MID code matches PT ID code (assigned at synthesis). Defined command opcodes are:

CMD	read/write of state ram bits
0	0-31
1	32-63
2	64-95
3	96-104

If AI (autoincrement)=1, a post increment of ram address is performed to simplify block read or write operation.

To perform test access, TAC is written and then TAD is written or read.

4.1.7 Test Data Register (TD)

Access : read/write

Reset Value : 00000000_H

31	
TEST DATA	

15	0
TEST DATA	

Refer to Test Command Register for a description of data.

A-1 Introduction to M256F Application

A-1.1 Overview

The important M256F characteristics are:

- 256 channels
- aggregate throughput of approximately 50 Mb/s @33Mhz clock
- HDLC, bit-synchronous PPP, octet-synchronous PPP, Transparent modes

These requirements can be met without changes to the PT core macro except for changes in constants e.g. M256F definitions for channel address bus width, channel command opcodes, and macro ID. An application shell adapts the SMIF interface to the Flexible Peripheral Interface (FPI) bus and the PT's simple interrupts to the M256F's interrupt vector bus. An interrupt buffer is implemented in the shell due to the latency of the M256F interrupt controller.

A-1.2 System Integration

The main PT interfaces are:

- FPI Slave Bus (PT SMIF adaptation in shell)
- Timeslot Assigner Transmit (TSAT) (no adaptation required)
- Transmit Buffer (TB) (no adaptation required)
- Interrupt Controller (vector and service latency adaptation in shell)

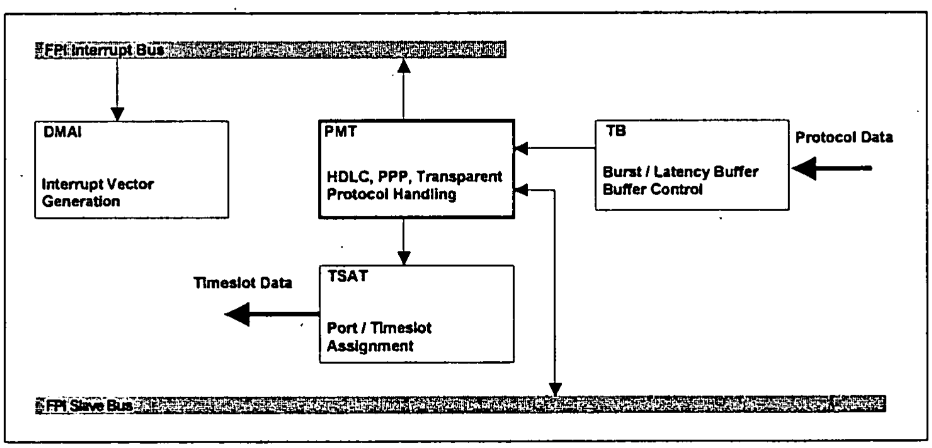

Figure A-1 System Integration in M256F

A-2 M256F PT Application Shell

The M256F application required adaptation of the PT in the following areas:

1. Configuration and control. Adapt SMIF to FPI bus which uses daisy chain concept to reduce bus area and avoid 3-state buses.
2. Interrupts. Adapt simple interrupt to a system conforming interrupt structure and daisy chain bus.

The reset also required adaption of the PT's single interrupt to separate hardware and software reset inputs. This was implemented transparently to the PT by using a project standard reset core.

The M256F uses a daisy chained bus structure for interrupts and configuration/control. In this concept, a macro's output bus becomes the input bus for the next macro in the chain. A macro forms its daisy chain output by 'or'ing, bit-by-bit, its output data with the input bus. To avoid disturbing other bus users, a macro holds its output data to all zeros until it is granted the bus.

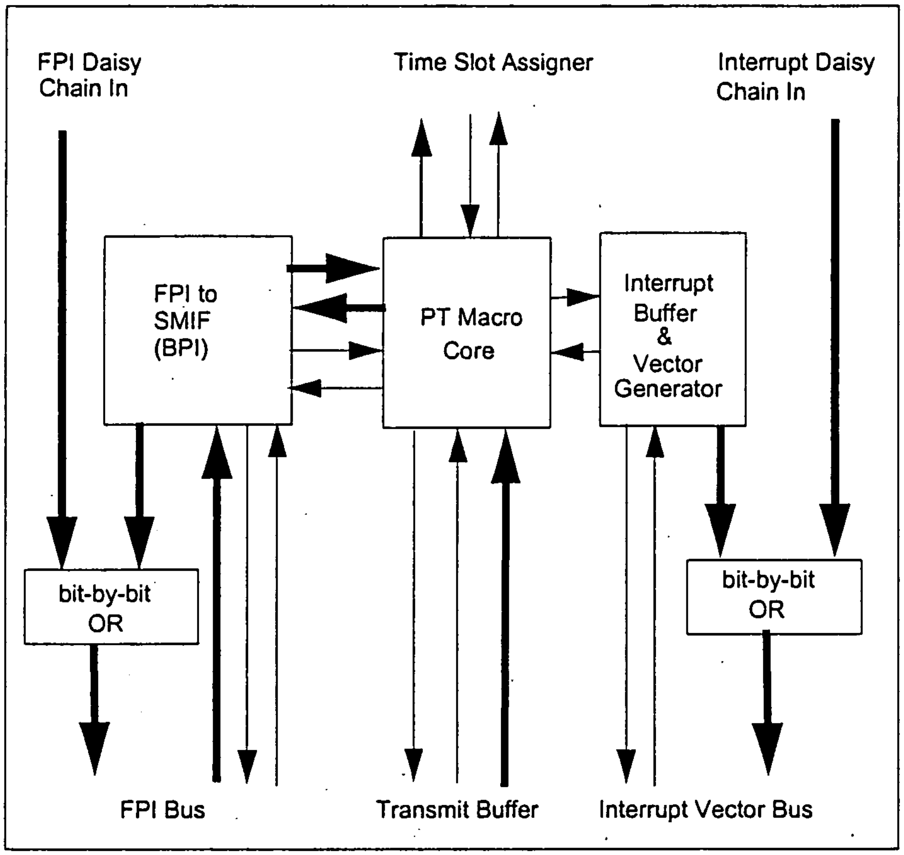

Figure A-2 Structure of M256F PT Application Shell

A-2.1 Transmit Buffer Adaptation

No shell adaptation other than renaming of signal names to satisfy M256F naming convention is required.

A-2.2 Time Slot Assigner Adaptation

No shell adaptation other than renaming of signal names to satisfy M256F naming convention is required.

A-2.3 Control and Configuration (SMIF) Adaptation

The M256F uses a simplified 32 bit variant of the FPI bus which supports overlapped read and write cycles only. FPI registers are dword aligned. The address bus does support octet addressing (bits 0 and 1 are not implemented).The data bus is daisy chained (see previous section).

The M256F uses a concept called the virtual register. The concept allows a data register which is distributed over more than one macro to be viewed by external software as a single register. This hides the internal macro function split which is not relevant to the user. Internally, the FPI bus interface decodes certain address ranges corresponding to these virtual data blocks. Three select signals are generated for PT:

1. Virtual Channel register select (PB_VC_TFPI_SEL_N). Data in this block is channel specific and is distributed over all macros handling protocol channels. PT captures channel configuration data located at several offset addresses in the channel register block. After a channel is initialized, PT returns this data when the channel's virtual channel register containing it is read.
2. Virtual Global register select (PB_VG_TFPI_SEL_N). Data in this block is also distributed over multiple macros but is not channel specific. This data group includes system interrupt queue ID, interrupt masks and test commands for PT. The test command register uses another level of addressing via a macro id field. When the register is written with a valid macro id, the ownership of the test command and test data register passes to that id. The PT is assigned "0010" for M256F.
3. PT specific register select (PB_PT_TFPI_SEL_N) for registers exclusive to the PT. Only the Extended Asynchronous Control Characters register is in this catagory.

The PT will not start an FPI cycle until the input ready signal PB_TFPI_RDY is asserted. Once a bus cycle begins, PT extends its read ready cycle i.e. drives its output data bus until the input PB_TFPI_RDY goes high. This signal is the "AND" combination of all macro ready signals and thus stretches a virtual register read until data from the slowest macro is available.

A-2.4 Interrupt Adaptation

The M256F maps interrupts into a 32 bit vector that is combined with interrupt vectors from other macros in a daisy chained bus. Each macro's interrupt vector output becomes the interrupt vector input of the next macro in the chain. The macro forms its output by the bit-by-bit "OR" of its own interrupt vector with the input interrupt vector. The macro must hold its interrupt vector to all zeros until it is granted the bus to avoid disturbing other macros which may be using the bus.

PT generates data UnderRun (UR) and Frame End (FE) interrupts for the channel. The interrupt vector includes a constant identifying the interrupt vector type, the 2 interrupt flags, the interrupt priority (queue ID), and the channel number.

The PT shell provides a 32 entry buffer for interrupt requests which is necessary to handle the interrupt controller's latency. In the worst case, PT can generate an interrupt every 5 clock cycles. Although the interrupt controller can handle this on average, the

Table A-1 PT Interrupt Vector

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
1	0	1	1	0	Queue ID (0-7)			0	0	0	0	0	0	0	0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
UR	FE	0	0	0	0	0	0	Channel Number(0-255)							

collisions with other bus users will delay service. The buffer stores the interrupt flags and channel number and generates the interrupt vector on the fly.

A-3 Interface Signals Description

All signals are active high until otherwise specified. Active low signals are designated by "_N" appended to their names.

Table A-2 Interfaces and Signal Description

Symbol name	I/O	Function
Clock and Reset		
SYSCLK	I	Internal clock (33 MHz) derived from FPI master bus
HW_RESET_N	I	Asynchronous reset of PT
SW_RESET_N	I	Software controlled reset of PT
Timeslot Assigner (TT) Interface		
TT_PT_RD_N	I	TT beginning read of channel protocol data (1 clock wide pulse). All TT inputs to PT must be valid during the time this signal is asserted. They will be sampled at the rising edge.
TT_PT_CH[7:0]	I	Logical channel number.
TT_PT_MASK[7:0]	I	Enable mask for PT_TT_D . Function of bit N (0-7) is: 0 => bit N is not used (set to '1' by PT). 1 => PT inserts protocol data in bit N.
TT_PT_SYNC	I	Synchronization for Transparent Mode SuperChannel
PT_TT_RDY	O	PT will complete transfer in current clock cycle
PT_TT_D[7:0]	O	Protocol Data as per TT_PT_MASK
PT_TT_CH_ON	O	Channel on/off status. 0=> channel is off. 1=> channel is on.
Transmit Buffer (TB) interface		
PT_TB_RD_N	O	PT read request for channel PT_TB_A service data.
PT_TB_A[7:0]	O	Channel number (0-255) for read request
TB_PT_D[31:0]	I	Channel data
TB_PT_RDY	I	TB will complete data transfer this clock

Table A-2 Interfaces and Signal Description (cont'd)

Symbol name	I/O	Function
TB_PT_STAT	I	Data type being read. Valid when TB_PT_RDY='1' and TB_PT_EMPTY='0'. 0 => 4 octets of service data. 1 => 0-3 octets of service data+ status octet
TB_PT_EMPTY	I	Transmit buffer empty flag. Valid when TB_PT_RDY='1' 0 => TB_PT_D is valid data/status. 1 => No data available for channel. TB_PT_D not valid.
Interrupt Controller Interface		
PT_IC_REQ_N	O	PT request for interrupt bus PT_IC_D 0 => PT requesting bus (active until IC_PT_GNT_N='0') 1 => PT not requesting bus
PT_IC_D[31:0]	O	Daisy chained interrupt bus output
IC_PT_GNT_N	I	Interrupt bus grant. 0 => PT granted PT_IC_D bus during next clock. 1 => PT not granted PT_IC_D. During next clock, PT_IC_D=IC_D.
IC_D[31:0]	I	Daisy chained interrupt bus input
Target FPI Slave Bus		
PB_TFPI_A[8:2]	I	Address bus.
PB_TFPI_D[31:0]	I	Daisy chained bus input data
PT_TFPI_D[31:0]	O	Daisy chained bus output data
TFPI_WR_N TFPI_RD_N	I I	Read/Write controls. Following codes are defined: WR_N = 1; RD_N = 1 => NOP WR_N = 0; RD_N = 1 => data written to DMUR WR_N = 1; RD_N = 0 => data read from DMUR
TFPI_VC_SEL_N	I	Virtual Channel register select.
TFPI_VG_SEL_N	I	Virtual Global register select.
TFPI_PT_SEL_N	I	PT macro specific register select.

Table A-2 Interfaces and Signal Description (cont'd)

Symbol name	I/O	Function
PT_TFPI_RDY	O	End of transfer indicator: 0 => PT not ready. Master must insert wait states 1 => end of transfer during this clock.
PB_TFPI_RDY	I	Ready enable. PT will not recognize *_SEL_N input signal or end its own ready cycle unless this signal is asserted high. 0 => delay start or completion of PT bus cycle. 1 => enable start or completion of PT FPI bus cycle.

A-4 Data Flow and Functional Timing

A-4.1 FPI Slave Bus

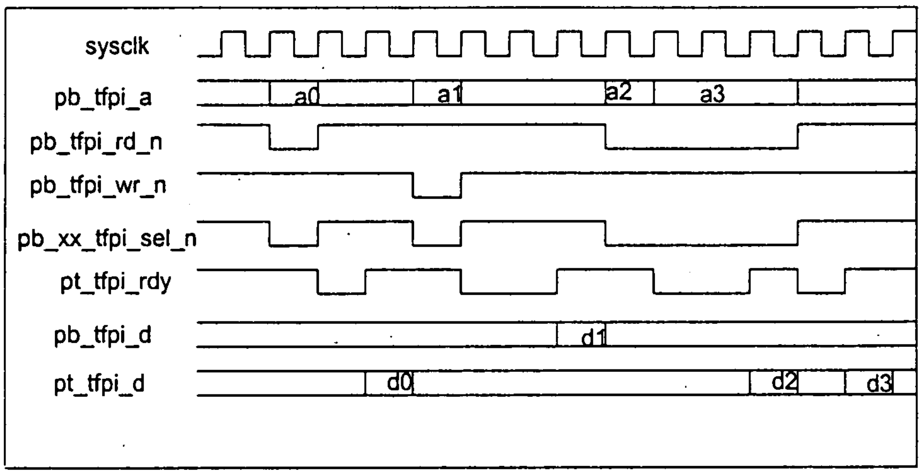

Figure A-3 FPI bus cycles : 1 read + 1 write + 2 overlapped reads

A-4.2 Interrupt Bus Interface

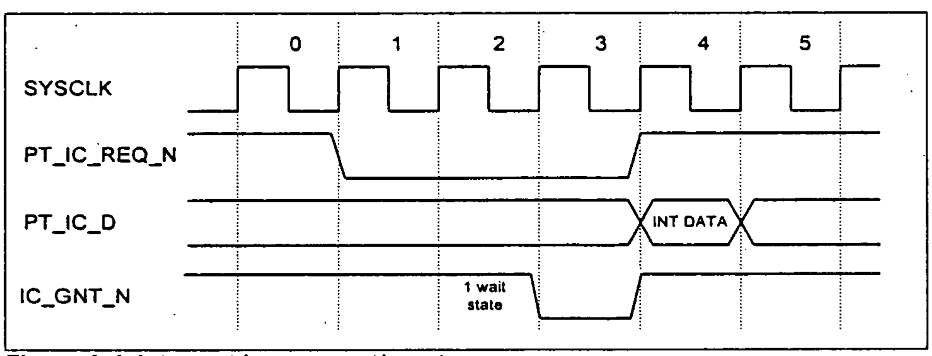

Figure A-4 Interrupt bus request/grant sequence

A-5 Register Description

A-5.1 Command Register

Access : read/write
Address : 00_H
Reset Value : 00000000_H

PB_VC_TFPI_SEL_N must be '0' to access this register.

31	16
CMD_XMIT(7:0)	don't care

15 8	7 0
don't care	CHAN(7:0)

Commands affect the Virtual Channel Registers (VCRs). When programming a channel, the command should be written only after all required parameters have been written to the VCRs. In the case of a debug command which reads a channel's programmed configuration, the command register is written first and then the VCR's address is read. For a debug read, PT will "or" data into the daisy chain bus only for those bits in the register it "owns" i.e. captured during programming phase. Otherwise it passes daisy chain input bus to the output unchanged

Transmit commands recognized by PT:

00000000 : No operation
00000001 : Init Channel. copies previouslly written VC registers to channel state ram.
00000010 : Channel off. PT turns off channel after clearing protocol pipeline.
00010000 : Debug. channel address accepted for virtual channel register read access.
00100000 : Transmit Idle. Update configuration data TFLAG. Send TFLAG (idle).
01000000 : Update. Alias for Init Channel command. Allows interworking with other macros that cannot execute Init Channel a 2nd time without disturbing the channel. PT requires this function to update FNUM.

The Transmit Idle is used to send PCM idle code. This command turns off the channel's TB interface and sends the Transparent mode idle flag TFLAG to the Time Slot Assigner (TT). Idle code will not be correctly sent unless TT mask is all "ones".

A-5.2 VC Channel Configuration Register

Access : read/write
Address :14_H
Reset Value : 00000000_H

PB_VC_TFPI_SEL_N must be '0' to access this register.

31	
FNUM(7:0)	TFLAG(7:0)

15											0
IFTF	0	FA	INV	TMP	0	CRC 32	CRC DIS	EACCM (3:0)	DEL	0	PMD(1:0)

Refer to layout of Channel Configuration in description of RB macro SMIF registers.

A-5.3 VC Asynchronous Control Character Map (ACCM) Register

Access : read/write
Address : 18_H
Reset Value : 00000000_H

PB_VC_TFPI_SEL_N must be '0' to access this register.

31															16
1F	1E	1D	1C	1B	1A	19	18	17	16	15	14	13	12	11	10

15															0
0F	0E	0D	0C	0B	0A	09	08	07	06	05	04	03	02	01	00

This map is used by channel for octet synchronous PPP mode only.When a bit is set, the corresponding character will be replaced by the Control ESC character in the outgoing data stream

A-5.4 VC Interrupt Queue ID (PT Interrupt Priority) Register

Access : read/write
Address : 20_H
Reset Value : 00000000_H

PB_VC_TFPI_SEL_N must be '0' to access this register.

31	
TQUEUE(2:0)	don't care

15	0
don't care	

TQUEUE: PT interrupt vectors will contain this parameter as their queue ID.

A-5.5 VC Channel Interrupt Masks Register

Access : read/write
Address : $2C_H$
Reset Value : 00000000_H

PB_VC_TFPI_SEL_N must be '0' to access this register

31	23	22	
Don't care	UE	FE	Don't care

15	0
Don't care	

PT nterrupt Masks:

0 => interupt is enabled.

1 => interrupt is disabled.

UE : UnderRun Interrupt Disable.

FE : Frame End Interrupt Disable

A-5.6 Extended Asynchronous Control Characters Register

Access : read/write

Address : 38_H

Reset Value : 00000000_H

PB_PT_TFPI_SEL_N must be '0' to access this register.

31	16
CHAR3(7:0)	CHAR2(7:0)

15	0
CHAR1(7:0)	CHAR0(7:0)

This register is only used by a channel in octet synchronous PPP mode. A character written to this register will be replaced with a Control Escape sequence when the corresponding enable flag is set in EACCM (3:0) field of channel configuration register.

A-5.7 Test Command and Address Register (TAC)

Access : read/write
Offset Address : 58_H
Reset Value : 00000000_H
PB_VG_TFPI_SEL_N must be '0' to access this register.

31

MID	0	0	0	AI	command

15 0

0	0	0	0	ADDRESS

MID : Macro ID Code. Must be 0010 to be accepted by PT.
AI : Auto Increment Function (enables post incress of ADDRESS following read or write access of Test Data Register.
Address : PT state ram address
Command : Specifies segment of PT state ram to read or write
Test registers are virtual global registers The function of test commands is explained in the RB macro description of the SMIF test register.

A-5.8 Test Data Register (TD)

Access : read/write
Address : $5C_H$
Reset Value : 00000000_H
PB_VG_TFPI_SEL_N must be '0' to access this register.

31

TEST DATA

15 0

TEST DATA

TD returns data read via TAC register.